

Esame di Calcolatori Elettronici T

23 Dicembre 2021 (Ing. Informatica)

Esercizio 1

Progettare un sistema, basato su un processore DLX dotato di **512 MB di EPROM** mappata agli indirizzi bassi e **3 GB di RAM** mappata agli indirizzi alti. Nel sistema sono presenti anche **due porte in input**, denominate **INPUT_A** e **INPUT_B** già progettate, ciascuna in grado di trasferire 8 bit mediante il protocollo di *handshake*. Ogni porta in input è connessa a un dispositivo esterno, che agisce in modo indipendente dall'altro, **senza la possibilità di poter modificare in alcun modo le connessioni tra porta e dispositivo esterno**.

Le due porte in input dovranno consentire al DLX la **lettura di 16 bit con un unico accesso nello spazio di indirizzamento** in accordo alla seguente strategia. Nel caso risulti che il **numero di trasferimenti (modulo 256)** a 16 bit eseguiti dal processore fino a quel momento sia **divisibile per 8**: INPUT_A fornirà gli 8 bit meno significativi (e INPUT_B gli 8 bit più significativi). **In caso contrario**, gli 8 bit meno significativi saranno forniti da INPUT_B (e gli 8 bit più significativi da INPUT_A). Si consideri 0 non divisibile per 8.

All'avvio, la **rete che tiene traccia del numero di trasferimenti (modulo 256)** eseguiti dovrà essere **inizializzata al valore 1Fh**. Una volta inizializzato il sistema, le operazioni definite in precedenza dovranno essere eseguite ininterrottamente e **unicamente mediante opportune reti logiche** (non saranno considerate valide soluzioni completamente software). I 16 bit, di tipo *signed*, letti contemporaneamente dal DLX dalle due porte dovranno essere memorizzati, come *word*, a **FFFF1000h** mediante opportune istruzioni software.

- Descrivere sinteticamente la soluzione indicando chiaramente quali sono i dispositivi utilizzati, gli indirizzi e i segnali di *chip-select*
- Progettare il sistema indicando le espressioni di decodifica e il range di indirizzi di tutte le periferiche, le memorie e i segnali e le connessioni di tutti i dispositivi con i bus di sistema. Evidenziare eventuali criticità.
- Scrivere il **codice necessario** assumendo che i registri da R20 a R25 non debbano essere ripristinati

Esercizio 2

- È possibile eseguire un'istruzione *jump and link* all'interno di un interrupt handler? Si o no?
- Motivare chiaramente e sinteticamente la risposta al punto precedente

Esercizio 3

- Nel DLX, sono presenti vincoli nell'accesso in memoria a word? Si o No?
- Motivare chiaramente e sinteticamente la risposta al punto precedente

Risposte vaghe e/o non focalizzate sulle domande del testo non saranno MINIMAMENTE considerate.

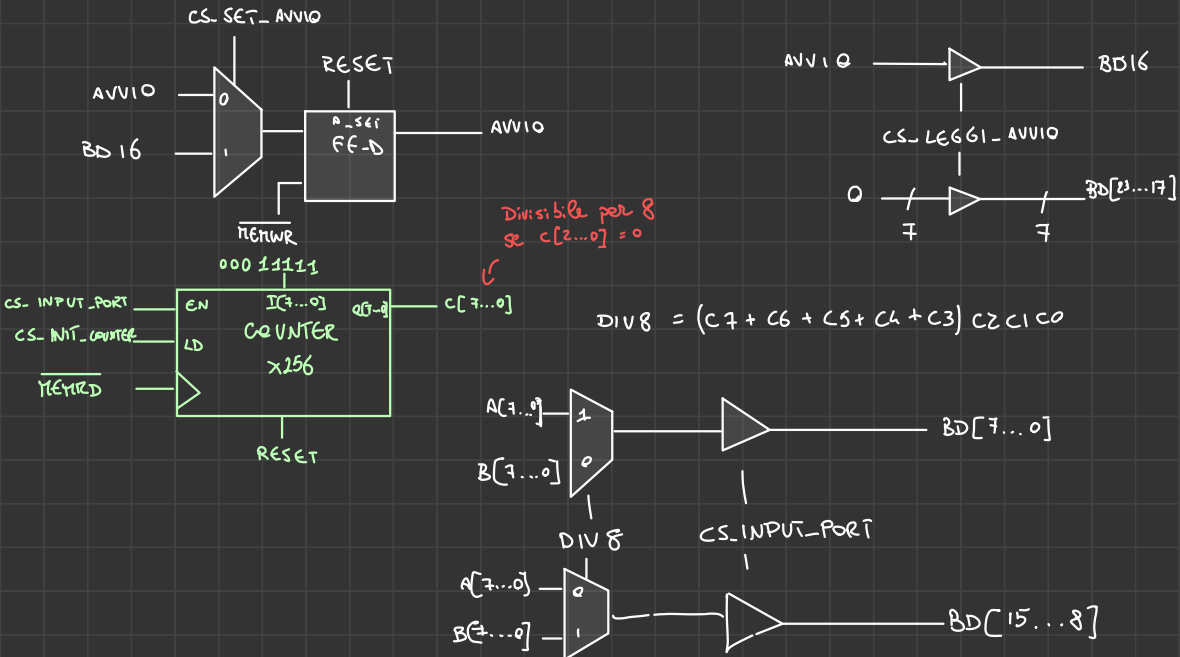
TAPPING

| | | | | | |
|----------------|--------------|----------------|-----------------|---------------|---------|
| 512KB EPROM | SET AVVIO | LEGGI AVVIO | INIT COUNTER | INPUT PORT | 3GB RAM |
|----------------|--------------|----------------|-----------------|---------------|---------|

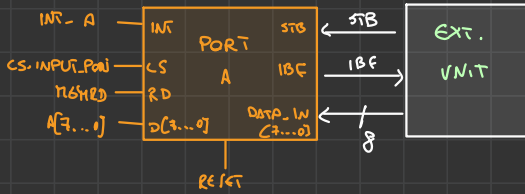
$$\begin{aligned}
 CS_EPROM_0 &= \overline{BA31} \overline{BA30} \overline{BA29} B60 \\
 \text{" } \quad \quad \quad 1 &= \text{" } \quad \quad \quad \text{" } B61 \\
 \text{" } \quad \quad \quad 2 &= \text{" } \quad \quad \quad \text{" } B62 \\
 \text{" } \quad \quad \quad 3 &= \text{" } \quad \quad \quad \text{" } B63
 \end{aligned}
 \quad
 \begin{aligned}
 CS_RAM_2GB_0 &= BA31 B60 \\
 \text{" } \quad \quad \quad 1 &= B61 \\
 \text{" } \quad \quad \quad 2 &= B62 \\
 \text{" } \quad \quad \quad 3 &= B63
 \end{aligned}$$

$$\begin{aligned}
 CS_RAM_512KB_0 &= \overline{BA31} \overline{BA30} \overline{BA29} B60 \\
 \text{" } \quad \quad \quad 1 &= \text{" } \quad \quad \quad \text{" } B61 \\
 \text{" } \quad \quad \quad 2 &= \text{" } \quad \quad \quad \text{" } B62 \\
 \text{" } \quad \quad \quad 3 &= \text{" } \quad \quad \quad \text{" } B63
 \end{aligned}
 \quad
 \begin{aligned}
 CS_512KB_0 &= \overline{BA31} \overline{BA30} \overline{BA29} B60 \\
 \text{" } \quad \quad \quad 1 &= \text{" } \quad \quad \quad \text{" } B61 \\
 \text{" } \quad \quad \quad 2 &= \text{" } \quad \quad \quad \text{" } B62 \\
 \text{" } \quad \quad \quad 3 &= \text{" } \quad \quad \quad \text{" } B63
 \end{aligned}$$

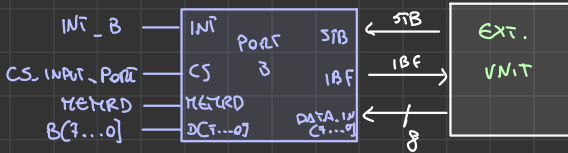
$$\begin{aligned}
 CS_INPUT_PORT &= \overline{BA31} \overline{BA30} BA29 B60 B61 \\
 CS_INIT_COUNTER &= \text{" } B63 \\
 CS_LEGGI_AVVIO &= \text{" } B62 \cdot \overline{RETURN} \\
 CS_SET_AVVIO &= \text{" } B62
 \end{aligned}$$



INPUT-PORT A,B



$$INT = INT_A \cdot INT_B \cdot \overline{AVVIO}$$



CODICE

```

000 0h    LHI R1, 0x2000;
000 4h    LBU R2, 0x002(R1);
000 8h    BEQZ R2, interrupt_handler;
000 ch    LB R3, 0x003(R1);
001 0h    SB R0, 0x002(R1);
001 4h    J main;
001 8h    interrupt_handler: LHI R4, 0xFFFF;
001 ch    LW R6, 0x000(R1);
002 0h    SW R5, 0x1000(R4);
002 4h    RfC;
002 8h
:
:
003 0h    main: ....      main = 0x0030
                                handler = 0x0018

```

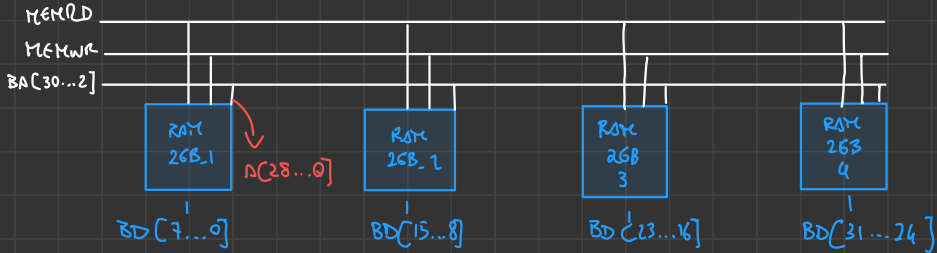
- Verificare se si è all'arrivo
- Se si è all'arrivo, porre arrivo a 0 e initialize counter

MEMORY

RAM 2 GB

$$2 \text{ GB} = 2048 \text{ MB} = 1024 \cdot 2048 \text{ KB} = 1024 \cdot 1024 \cdot 2048 \text{ B} = \\ = 2 \cdot 1024^3 = 2^{31}$$

BA : 30...0 \rightarrow 30...2



RAM 512 MB e EPROM 512 MB

$$512 \text{ MB} = 1024^2 \cdot 512 \text{ Byte} = 10^{20} \cdot 10^9 = 10^{29}$$

BA[28...0] \rightarrow A[28...0]